

ОЦІНКА СКЛАДНОСТІ ПРОГРАМ ЗА ДОПОМОГОЮ МЕТРИКИ ХОЛСТЕДА

Проблема оцінки якості створених програмних засобів завжди була і є дуже важливою задачею, рішення якої дозволяє на ранніх стадіях життєвого циклу визначити характеристики програми, яка створюється. Існує ряд методик, які направлені на оцінку параметрів майбутніх програм, що дозволяє адекватно оцінити трудовитрати і вартість розробки. Серед них представлено метрику Холстеда, яка оцінює характеристики програмних засобів на основі лексичного аналізу тексту програм.

Метрика Холстеда використовується для підрахунку характеристик розміру програм. Ця метрика відноситься до метрик, які обчислюються на основі аналізу рядків і синтаксичних елементів вихідного коду програми.

Основу метрики Холстеда складають шість базових характеристик програми, які вимірюють:

- число унікальних операторів (словник операторів) n_1 ;
- число унікальних операндів (словник операндів) n_2 ;
- загальне число операторів в програмі N_1 ;
- загальне число операндів в програмі N_2 ;
- число входжень j -го оператора, $j = 1, 2, \dots, n_2$
- число входжень j -го операнда, $j = 1, 2, \dots, n_2$;

На основі цих характеристик знаходять:

- 1) довідник: $n = n_1 + n_2$;
- 2) довжина програми: $N = N_1 + N_2$;
- 3) оціночну довжину програми: $\tilde{N} = n_1 \cdot \log_2 \cdot n_1 + n_2 \cdot \log_2 \cdot n_2$;
- 4) потенційний об'єм програми: $V^* = (2 + n_2^*) \cdot \log_2 \cdot (2 + n_2^*)$;
- 5) складність програми: $D = (n_1 / 2) \cdot (N_2 / n_2)$;
- 6) рівень програми: $L = 1 / D$;
- 7) зусилля на розробку програми: $E = V / L$;
- 8) кількість помилок в програмі: $B = V / 3000 = E^{2/3} / 3000$;
- 9) час розробки програми: $T = E / S$,

де $S = 18$ – число Страунда (кількість елементарних розумових операцій в секунду).

Крім свого прямого призначення метрики довжини програм і довжини реалізації можна використовувати для виявлення недосконалості програмування, які є наслідком застосування не зовсім вдалих прийомів програмування. Якщо розрахункові значення довжини програми і довжини реалізації відрізняються більш чим на 10%, то це свідчить про можливу наявність в програмі наступних шести класів недосконалості:

1. Наявність послідовності доповнюючих один одного операторів до одного і того ж операнда. В цьому випадку буде виконано абсолютно непотрібні дії.
2. Наявність неоднозначних операндів.
3. Наявність синонімічних операндів.
4. Наявність загальних підвиразів, що призводить до додаткового часу виконання програми.
5. Непотрібне присвоювання, що може призвести до збільшення об'єму пам'яті, яка резервується під змінну програми, і збільшує розмір довідника.
6. Наявність виразів, які не представлені в згорнутому вигляді як визначник множника.

Таким чином, довжина реалізації і об'єм програми визначаються на основі аналізу тексту програми шляхом підрахунку кількості операндів і операторів, а також числа їх входжень в текст програми, тобто на основі лексичного аналізу тексту програм.

Співвідношення Холстеда дозволяє контролювати процес розробки програмних засобів, якщо заздалегідь визначити їх довжину, то можна виходити на цю задану довжину модулів при проектуванні програм.

Отже, розрахункові метричні характеристики (довжина модулів, їх число, кількість ієрархічних рівнів) задають оптимальні параметри структури програмних засобів, найбільш раціональні в аспекті забезпечення якості реалізації проекту.